MULTIMEDIA UNIVERSITY

STUDENT ID NO

MULTIMEDIA UNIVERSITY

FINAL EXAMINATION

TRIMESTER 1, 2018/2019

DCS5088 – OBJECT ORIENTED PROGRAMMING

(For DIT students only)

22 OCTOBER 2018
2.30 p.m – 4.30 p.m
(2 Hours)

INSTRUCTIONS TO STUDENTS

1. This examination paper consists of **13** pages excluding cover page.
2. There are **4** structured questions.
3. Please write all your answers in the Answer Booklet provided.

QUESTION 1 (30 Marks)

a) Write a complete program to calculate the total price of a holiday package based on the selected package and number of adults and children input by the user. Use the following requirements and sample input and output to write the program.

(10 marks)

- Use *do...while* statement to prompt the user for correct package selection. If the input is incorrect display *"Invalid selection"* and prompt again for package selection.

- Use *switch* statement to determine the price based on the available package.

Package	Price (RM)
A	Adult = 1000.00 Child = 500.00
B	Adult = 800.00 Child = 600.00
C	Adult = 500.00 Child = 300.00

- Use *if* statement to deduct 20% from the total price if the price is greater than RM3000.00.

Sample input and output :

```
Select travel package <A,B,C> : L
Invalid selection.

Select travel package <A,B,C> : A

Enter no of adult : 4
Enter no of children : 3

Total price : RM4400.00
```

Continued...

b) Write the output for the following code segment. (5 marks)

```
char subject[20] = "Programming";
char *p1;
int a = 2, b = 4, *p2, *p3;
p1 = subject;
cout<<*p1<<endl;
cout<<*(p1+3)<<endl;
p2 = &a;
p3 = &b;
*p2 += *p3;
cout<<a<<endl;
*p3 = ++b;
cout<<b<<endl;
cout<<*p2+a<<endl;
```

c) Write the answers to all of the following questions on function based on the description given. You may create variables whenever necessary.

i. Write a void function prototype for a function named *Multiply(...)* that accept two integers from *main( )* and display the multiplication result of the two integers in its definition. (2 marks)

ii. Write a function prototype for the following function call.

```
float pay = CalculatePay (&salary, &deduction);
```

where variables *pay*, *salary*, and *deduction* are floating point values. (2 marks)

iii. Write two variable initializations and a function call for the following function definition. (2 marks)

```
void HighestResult (int &quiz1, int &quiz2)
{
      int highest=quiz1;
      if(quiz2>highest)
           highest = quiz2;
      cout<<"Highest quiz mark : "<<highest<<endl;
}
```

Continued...

d) Modify the following program based on the sample input and output and requirements given below :

i. For the *Input(...)* function :
- Create a mutator function to set the type and price of the guitar based on the values passed from *main( )* function.

ii. In *main( )* function :
- Declare two variables to store guitar type and guitar price.
- In the for loop of three times, prompt for guitar type and prompt for guitar price and call function *Input(...)* and passed two arguments which are the guitar type and guitar price.

*(Note : Write codes for parts labelled **i** and **ii** only)* (9 marks)

Program :

```
#include<iostream>
#include<iomanip>
using namespace std;
class Guitar
{
      private :
         string type;
         float price;
      public :
         void Input()
         {
              cout<<"Enter guitar type : ";
              cin>>type;
              cout<<"Enter price : ";
              cin>>price;
              cout<<endl;
         }
        void Display()
        {
            cout<<setw(10)<<type<<setw(8)<<price<<endl;
        }
};

int main()
{
      Guitar g[3];
      int i;
      for(i=0;i<3;i++)
          g[i].Input();
      cout<<endl;
      cout<<setw(10)<<"Type"<<setw(8)<<"Price"<<endl;
      for(i=0;i<3;i++)
        g[i].Display( );
      return 0;
}
```

i

ii

Continued...

Sample input and output :

```
Enter guitar type : Acoustic
Enter price : 290.00

Enter guitar type : Electric
Enter price : 849.00

Enter guitar type : Stratocaster
Enter price : 551.00

            Type                 Price
        Acoustic                290.00
        Electric                849.00
    Stratocaster                551.00
```

Continued...

QUESTION 2 (20 Marks)

a) The following program supposed to convert temperature in Celsius into Fahrenheit. However, the program has errors. Find the error line number and write the correct code. (9 marks)

```
#include(iostream)                                                    //Line 1
using namespace std;                                                  //Line 2
                                                                      //Line 3
class TempConvert{                                                    //Line 4
    private :                                                         //Line 5
      float celsius, fahrenheit;                                      //Line 6
                                                                      //Line 7
    public :                                                          //Line 8
      TempConvert(){                                                  //Line 9
      cout<<c<<" Celsius = "<<f<<" Fahrenheit."; }                    //Line 10
                                                                      //Line 11
      void Input(){                                                   //Line 12
      cout<<"\nEnter temperature in Celsius : ";                      //Line 13
      cin>>celsius; }                                                 //Line 14
                                                                      //Line 15
      friend float(TempConvert);                                      //Line 16
                                                                      //Line 17
      void Display(float tempF){                                      //Line 18
      cout<<celsius<<" Celsius = "<<tempF<<" Fahrenheit."}            //Line 19
                                                                      //Line 20
       TempConvert~(){                                                //Line 21
       cout<<"\nEnd of conversion."<<endl; }                          //Line 22
};                                                                    //Line 23
                                                                      //Line 24
float convert(TempConvert temp)                                       //Line 25
{                                                                     //Line 26
      fahrenheit = celsius * 1.8 + 32;                                //Line 27
      return temp.fahrenheit;                                         //Line 28
}                                                                     //Line 29
                                                                      //Line 30
int main()                                                            //Line 31
{                                                                     //Line 32
      TempConvert temp(0,32);                                         //Line 33
      float tempF;                                                    //Line 34
      temp.Input();                                                   //Line 35
      tempF = convert();                                              //Line 36
      temp.Display();                                                 //Line 37
      return 0;                                                       //Line 38
}                                                                     //Line 39
```

Solution Example :

Line 1	#include<iostream>

Continued...

b) Write the answers for all of the following questions based on the program given below : (7 marks)

```
#include<iostream>
using namespace std;

class Cake {
      private :
      string type;
      float quantity;
      public:
      void Purchase(string t, float q)
      {
            type = t;
            quantity = q;
      }
      string getType()
      { return type; }

      int getQuantity()
      { return quantity; }
};


int main()
{
    Cake *ck;
    int noCake, i, qty;
    string cakeType;

    cout<<"Enter how many type of cake : ";
    cin>>noCake;

    _________Q2b(i)_____________

    for(i=0;i<noCake;i++)
    {
      cout<<"\nEnter type of cake <fondant, sponge, meringue> : ";
      cin>>cakeType;
      cout<<"Enter quantity : ";
      cin>>qty;
    ___________Q2b(ii)_____________
    }

    cout<<"\nYou have purchased : "<<endl;
    for(i=0;i<noCake;i++)
       ___________Q2b(iii)_____________________

      return 0;
}
```

Continued...

i. At *labelled Q2b(i),* declare a dynamic array of with size equal to value input by user in at *noCake* variable.

ii. At *labelled Q2b(ii),* write code segment to call object's member function *Purchase(...)* with two arguments : cake type (*cakeType*) and quantity (*qty*). Use arrow operator (- >) to access object's members.

iii. At *labelled Q2b(iii),* write code segment to display quantity for each type of cake purchase by user. Use indirection (*) and dot (.) operator to perform function call to *getQuantity( )* and *getType( )* functions.

c) Create a derived class named **3D** for the base class named **Shape** based on the code given below.

- class **3D** inherits publicly from class **Shape**.
- private data members : *depth* (int)
- public member functions :
  - *void SetDepth (int dpth)* – set the function parameter dpth to depth
  - *void CalculateVolume ( )* – display the result of height * length * depth

(4 marks)

```
class Shape{
    private :
        int height, length;

    public:
        void SetHeight (int hgt)
        {height = hgt ;}

        void SetLength (int lth)
        {length = lth;}
};
```

Continued...

QUESTION 3 (20 Marks)

a) Write a complete program based on the following class declaration for **Calculate**. Use the following requirements and sample input and output to write the program. (12 marks)

i. Create class **Trapezium.**
- private data members : *perimeter*(int)
- public member functions :
  - For the overloaded *Calculate operator+(...)* function :
    - Declare a constant **Trapezium** reference object as parameter.
    - Return a **Calculate** object whose total is addition of data member perimeter and function parameter perimeter.
  - In *void SetPerimeter( )* function :
    - Prompt user for trapezium perimeter.

ii. In *main( )* function :
- Create two objects of **Trapezium** class named *A* and *B*.
- Call *SetPerimeter( )* function for each object, A and B.
- Declare a **Calculate** object named *C*.
- Assign value to object *C* with the result of *A's* perimeter added with *B's* perimeter by calling the **Calculate** overloaded operator addition function.
- Display *C* details by calling *Display( )* function.

```
#include<iostream>
#include<iomanip>
using namespace std;
class Calculate{
  private :
      int total;
      float price_per_cm;
  public :
      Calculate(int ttl)
      {
       total = ttl;
       price_per_cm = 2.00;
      }

      void Display()
      {
       cout<<"Perimeter (cm) : "<<total<<endl;
       cout<<"Ribbon price per cm(RM) : "<<fixed
        <<setprecision(2)<<price_per_cm<<endl;
       cout<<"Total ribbon price (RM) : "<<fixed
        <<setprecision(2)<<total*price_per_cm<<endl;
      }
};
```

Continued...

Sample input and output :

```
Enter trapezium perimeter (cm) : 20
Enter trapezium perimeter (cm) : 30
Perimeter (cm) : 50
Ribbon price per cm (RM) : 2.00
Total ribbon price (RM) : 100.00
```

b) Study the following program. Assume *ticket price* entered by user is RM 100 and *tax* is 0.2. (8 marks)

i. Write the output when virtual keyword is included in the program.
ii. Write the output when virtual keyword is removed from the program.

```
#include<iostream>
#include<iomanip>
using namespace std;

class Price{
      protected :
         float price, tax;
      public :
         void SetData(float p, float t){
           price = p;
           tax = t; }
         virtual void Pay(){
           cout<<"No calculation for ticket price.\n"; }
};

class Adult : public Price
{
      public :
         void Pay(){
           cout<<"Adult price is RM "<<price+(tax*price)<<endl; }
};

class Child : public Price
{
      public :
         void Pay(){
           cout<<"Child price is RM "<<(price/2)+(tax*price)<<endl;}
};
```

Continued...

```
int main()
{
    float price, tax;
    Price *p;
    p = new Price;
    Adult *a;
    a = new Adult;
    Child *c;
    c = new Child;
    cout<<"Enter ticket price (RM) : ";
    cin>>price;
    cout<<"Enter tax : ";
    cin>>tax;
    p = a;
    p->SetData(price,tax);
    p->Pay();
    p = c;
    p->SetData(price,tax);
    p->Pay();
    return 0;
}
```

Continued...

QUESTION 4 (30 Marks)

Write a complete program that gets user input for the sales of three premium bags for the first four months of the year. The program will display a summary showing the total and average sales for each bag.

- Declare a global constant variable for the program:
  - *SIZE* (int) : Set the constant value to 4. This value is for the size of the array *month* used in this program.

- Create a class called **Bag**:
  - Protected data members :
    - *type* : string
    - *material* : string
  - Public member functions:
    - *SetBag(....)* : Contains two string parameters to set the *type* and *material.*
    - *SetSales( )* : Set it as a pure virtual function.

- Create a class called **DesignerBag**[derived publicly from the class **Bag**]
  - Private data members :
    - *name* : string
    - *month* : string[SIZE]
    - *sales* : float[SIZE]
    - *total* : float
  - Public member functions:
    - Parameterized constructor :
      - Contains three parameters of string type. One of the parameters is used to set the *name* and the other two parameters will be passed to function *SetBag(...)* to set the *type* and *material.*
      - Set the array *month* with values "Jan", "Feb", "Mac", "Apr".
      - Set the *total* to 0.
    - *SetSales( ) :*
      - Get user input for *sales* array of four months.
      - Accumulate the *sales* array element in *total.*
    - *Display( )*
      - Display the *name, type, material* and *total.*
    - *Average(DesignerBag&)*:
      - Set this function as a *friend* of the class.
      - Refer to the instructions after this to define the function.

- Create a function called *Average(DesignerBag&)* outside the class.
  - Parameter : Reference object of **DesignerBag** class.
  - Returns the average of the *total* (data member of **DesignerBag** class).

Continued...

- In *main()*:
  - Declare a pointer object, *z* of **DesignerBag** class.
    - Use this pointer to create a new dynamic object. Pass the values "Zucci", "Sling", "Leather" that will set the object's *name*, *type* and *material*.
    - Call *SetSales()*.
  - Declare a pointer object, *h* of **DesignerBag** class.
    - Use this pointer to create a new dynamic object. Pass the values "Hemres", "Clutch", "Polyurethane" that will set the object's *name*, *type* and *material*.
    - Call *SetSales()*.
  - Declare a pointer object, *s* of **DesignerBag** class.
    - Use this pointer to create a new dynamic object. Pass the values "Shanel", "Tote", "Synthetic" that will set the object's *name*, *type* and *material*.
    - Call *SetSales()*.
  - For each dynamic object (*z, h, s*).
    - Call *Display()*
    - Display the average of the sales by calling *Average(....)* by passing the dynamic object using dereference operator.
  - Deallocate memory for all the dynamic objects.

[Note: Refer to sample output given below. The *bold items* are inputs entered by user.]

Sample input and output :

```
--------------------------------------------------------------------------
       Sales Data Entry for Zucci Sling Bag
--------------------------------------------------------------------------

Enter Jan sales: 10000.00
Enter Feb sales: 15000.00
Enter Mac sales: 20000.00
Enter Apr sales: 10500.00

--------------------------------------------------------------------------
        Sales Data Entry for Hemres Clutch Bag
--------------------------------------------------------------------------

Enter Jan sales: 20000.00
Enter Feb sales: 12000.00
Enter Mac sales: 25000.00
Enter Apr sales: 22000.00


--------------------------------------------------------------------------
       Sales Data Entry for Shanel Tote Bag
--------------------------------------------------------------------------

Enter Jan sales: 25000.00
Enter Feb sales: 15000.00
Enter Mac sales: 10000.00
Enter Apr sales: 13000.00

--------------------------------------------------------------------------
```

Continued...

Sales Report for Quarter Year 2018

Name	Type	Material	Total Sales(RM)	Average(RM)
Zucci	Sling	Leather	55500.00	13875.00
Hemres	Clutch	Polyurethane	79000.00	19750.00
Shanel	Tote	Synthetic	63000.00	15750.00

End of Page.